CLAIMS:

I claim:

1. A server page configured for processing by a server page engine, the server page comprising:

at least one markup language fragment defining a user interface for a first view;

an additional markup language fragment defining a link to a second view; and,

a custom tag conditionally including said additional markup language fragment only if a role detected for an end user attempting to access the first view also has been defined in a deployment descriptor as an authorized role for accessing said second view.

2. The server page of claim 1, wherein said first and second views are Java server pages (JSPs) and wherein said deployment descriptor is a configuration file for an application framework incorporating said JSPs.

3. The server page of claim 2, wherein said application framework is the Struts framework.

4. A system for programmatic role-based security in a dynamically generated user interface, the system comprising:

an application framework configured through a deployment descriptor comprising a listing of a set of views, a listing of associated program logic and a listing of a set of authorized roles for selected ones of said views;

a first view listed in said deployment descriptor and comprising a linkage to a second view listed in said deployment descriptor; and,

access checking logic disposed in said first view and programmed to omit said linkage where a role of an end user accessing said first view is not authorized to access said second view according to said listing of said set of authorized roles in said deployment descriptor.

5. The system of claim 4, wherein said application framework comprises the Struts framework.

6. The system of claim 4, wherein said program logic comprises servlets and wherein said views comprise Java server pages (JSPs).

7. The system of claim 6, further comprising a custom tag disposed in said first view for invoking said access checking logic and for omitting said linkage responsive to said access checking logic.

8. A method for programmatic role-based security in a dynamically generated user interface, the method comprising the steps of:

authenticating access to a rendering of a selected view based upon a role of an end user requesting access to said selected view;

processing said selected view to identify a method call to access checking logic;

comparing said role to a set of roles authorized to access a different view associated with said access checking logic; and,

disposing a link to said different view in said rendering of said selected view if said role matches a role in said set.

9. The method of claim 8, wherein said step of authenticating comprises the step of comparing said to a set of roles associated with said selected view to locate a match for said role.

10. The method of claim 9, wherein said locating step comprises the step of parsing a deployment descriptor for an application framework hosting said selected view and said different view to identify said set of roles.

11. The method of claim 8, wherein said processing step comprises the step of invoking external access checking logic for a located server page tag referencing said access checking logic.

12. A machine readable storage having stored thereon a computer program for programmatic role-based security in a dynamically generated user interface, the computer program comprising a routine set of instructions which when executed by a machine cause the machine to perform the steps of:

authenticating access to a rendering of a selected view based upon a role of an end user requesting access to said selected view;

processing said selected view to identify a method call to access checking logic;

comparing said role to a set of roles authorized to access a different view associated with said access checking logic; and,

disposing a link to said different view in said rendering of said selected view if said role matches a role in said set.

13. The machine readable storage of claim 12, wherein said step of authenticating comprises the step of comparing said to a set of roles associated with said selected view to locate a match for said role.

14. The machine readable storage of claim 13, wherein said locating step comprises the step of parsing a deployment descriptor for an application framework hosting said selected view and said different view to identify said set of roles.

15. The machine readable storage of claim 12, wherein said processing step comprises the step of invoking external access checking logic for a located server page tag referencing said access checking logic.

16. A method for programmatic role-based security in a dynamically generated user interface, the method comprising the steps of:

configuring a deployment descriptor to specify a set of roles authorized to access renderings of different views in a distributable application;

programming external access checking logic to match a parameterized role with a role disposed in said set of roles in said deployment descriptor; and,

composing a server page to include a reference to said external access checking logic and to invoke said external access in order to conditionally incorporate a link to a specific view associated with a specific set of authorized roles.